



Full credit is given to the above companies including the OS that this PDF file was generated!

Linux Ubuntu 26.04 Manual Pages on command 'wasm-interp.1'

\$ man wasm-interp.1

WABT(1) General Commands Manual WABT(1)

NAME

wasm-interp ? decode and run a WebAssembly binary file

SYNOPSIS

wasm-interp [options] file

DESCRIPTION

wasm-interp Read a file in the wasm binary format, and run it in a stack-based interpreter.

The options are as follows:

--help Print a help message

--version

Print version information

-v, --verbose

Use multiple times for more info

--enable-exceptions

Enable Experimental exception handling

--disable-mutable-globals

Disable Import/export mutable globals

--disable-saturating-float-to-int

Disable Saturating float-to-int operators

--disable-sign-extension

Disable Sign-extension operators

--disable-simd

Disable SIMD support

--enable-threads

Enable Threading support

--enable-function-references

Enable Typed function references

--disable-multi-value

Disable Multi-value

--enable-tail-call

Enable Tail-call support

--disable-bulk-memory

Disable Bulk-memory operations

--disable-reference-types

Disable Reference types (externref)

--enable-annotations

Enable Custom annotation syntax

--enable-code-metadata

Enable Code metadata

--enable-gc

Enable Garbage collection

--enable-memory64

Enable 64-bit memory

--enable-multi-memory

Enable Multi-memory

--enable-extended-const

Enable Extended constant expressions

--enable-all

Enable all features

-V, --value-stack-size=SIZE

Size in elements of the value stack

-C, --call-stack-size=SIZE

Size in elements of the call stack

-t, --trace

Trace execution

--wasi Assume input module is WASI compliant (Export WASI API the the module and invoke

_start function)

-e, --env=ENV

Pass the given environment string in the WASI runtime

-d, --dir=DIR

Pass the given directory to the WASI runtime

--run-all-exports

Run all the exported functions, in order. Useful for testing

--host-print

Include an importable function named "host.print" for printing to stdout

--dummy-import-func

Provide a dummy implementation of all imported functions. The function will log the call and return an appropriate zero value.

EXAMPLES

Parse binary file test.wasm, and type-check it

```
$ wasm-interp test.wasm
```

Parse test.wasm and run all its exported functions

```
$ wasm-interp test.wasm --run-all-exports
```

Parse test.wasm, run the exported functions and trace the output

```
$ wasm-interp test.wasm --run-all-exports --trace
```

Parse test.wasm and run all its exported functions, setting the value stack size to 100 elements

```
$ wasm-interp test.wasm -V 100 --run-all-exports
```

SEE ALSO

wasm-decompile(1), wasm-objdump(1), wasm-stats(1), wasm-strip(1), wasm-validate(1),
wasm2c(1), wasm2wat(1), wast2json(1), wat-desugar(1), wat2wasm(1), spectest-interp(1)

BUGS

If you find a bug, please report it at

<https://github.com/WebAssembly/wabt/issues>.

Debian

\$Mdocdate\$

WABT(1)